REMARKS

Claims 1-44 are currently pending. Claims 1-44 were rejected.

Claim Rejections – 35 USC § 102(e)

The Examiner rejected claims 1-4, 6-16, 18, 22-25, 27-36, 39, and 43-44 as being anticipated by Christie, U.S. Patent No. 6,877,084.

Claim 1 recites:

> A digital data processor comprising an instruction unit, said instruction unit comprising a **code page** that is partitioned for storing in a first section thereof a plurality of instruction words and, in association with at least one instruction word, in a second section thereof an extension to said at least one **instruction word**.

The Patent Office has asserted (page 5, lines 1-7, of the Final Office Action dated January 29, 2007) as follows:

> As per claim 1, Christie discloses a digital data processor (See column 4, lines 59-63) comprising an instruction unit (See column 5, lines 34-44), said instruction unit comprising a code page (See column 7, lines 36-45: Ability to page indicates a code page) that is partitioned for storing in a first section thereof a plurality of instruction words (See figure 7: Standard Register Set 84) and, in association with at least one instruction word, in a second section thereof an extension to said at least one instruction word (See figure 7: Extended Register Set).

Christie, in Figure 4, shows an embodiment of a computer system including a central processing unit 32 and a memory 36. Memory 36 provides instructions to the CPU 32 through north bridge 34. The CPU 32 executes the instructions (column 5, lines 13-23). "Instructions and data reside at addresses within memory 36…" (column 5, lines 16-17). Christie discloses "CPU 32 includes an internal cache memory (i.e., an internal cache) configured to store instructions and data previously accessed by CPU 32" (column 5, lines 24-26). As shown in Figure 5, Christie's CPU 32 includes an execution core 52 that is coupled separately to instruction cache 50, register file 60, and data cache 54.

Christie, column 7, lines 46-67, discloses as follows:

> Instruction cache 50 is a high speed cache memory for storing instructions. Execution core 52 fetches instructions from instruction cache 50 for execution. Instruction-cache 50 may employ any suitable cache organization, including direct-mapped, set associative, and fully associative configurations. If a requested instruction is not present in instruction cache 50, instruction cache 50 may communicate with interface unit 56 to obtain the requested instruction. Such communication may result in the filling/replacement of a cache line in instruction cache 50. Additionally, instruction cache 50 may communicate with the memory management unit to receive physical address translations for virtual addresses fetched from instruction cache 50.
>
> Execution core 52 executes instructions fetched from instruction cache 50. Execution core 52 obtains register operands from register file 60, and stores register result values within register file 60. The size of operands is dependent upon the operating mode of CPU 32, and may be overridden by instructions as described below. Execution core 52 obtains memory operands from data cache 54, and provides memory result values to data cache 54 as described below.

Christie discloses, column 8, lines 39-40, as follows: "Decode unit 76 detects any register source operand references in instructions and requests the operand values from register file 60."

Christie, column 9, lines 30-42, discloses as follows:

> When CPU 32 (FIG. 5) is operating in REX32 mode and instruction 80 identifies a register source operand but does not include optional prefix portion 82, decode unit 76 requests the contents of a corresponding standard register of standard register set 84 from register file 60. The execution of instruction 80 is thus carried out using the standard register from standard register set 84 of register file 60. On the other hand, when CPU 32 is operating in REX32 mode and instruction 80 includes optional prefix portion 82 and references a register source operand, decode unit 76 requests the contents of a corresponding extended register of extended register set 86 from register file 60. The execution of instruction 80 is thus carried out using the extended register from extended register set 86 of register file 60.

Christie discloses a register file that contains operands and not instruction words. An instruction word is a computer word containing an instruction rather than data - - http://www.answers.com/instruction%20word. An instruction word includes an opcode. Christie

does not disclose the standard register set 84 and extended register set 86 store instruction words and extensions to the instruction words, respectively. In fact, the registers of Christie do not store instruction words and portions of the instruction words are used to selected one of the standard or extended registers (column 9, lines 19-20). In addition, while the registers of the extended register set extend the number of registers for use by the CPU, none of the extended registers form an extension to any one register in the standard register set. Christie discloses decode unit 76, which is not part of the register file 60, detects any register source operand references in instructions and requests the operand values from register file 60 (column 8, lines 39-41). As register file 60 only stores operand values and not opcodes, register file 60 does not store instruction words because it does not store opcodes as well as operands and instructions are formed of a command section called an opcode and at least one value called an operand.

Contrary to the Patent Office assertions, on page 5, lines 4-7, of the Final Office Action dated January 29, 2007, register sets 84, 86 do not form a memory page or code page. Christie discloses register sets 84, 86 comprise a register file 60. Instruction words are stored in instruction cache 50, but operands, not instruction words, are stored in the register file 60. The register file 60 the Patent Office has used as a basis of rejection does not comport to a code page and does not comport to the storage of instruction words.

Thus, claim 1 is not anticipated by Christie. As independent claims 22 and 43 recite elements similar to those discussed above with reference to claim 1, they are likewise in condition for allowance. As all of claims 2-4, 6-16, 18, 23-25, 27-36, 39, and 44 depend upon claims 1, 22 and 43, they are likewise in condition for allowance.

Why does the Patent Office discuss Yates with respect to claim 44 when claim 44 has been rejected by the Patent Office as being anticipated by Christie?

Claim Rejections – 35 USC § 103(a)

The Examiner rejected claims 5, 17, 19-21, 26, 38 and 40-42 as being unpatentable over Christie in view of Yates et al., U.S. Patent No. 6,397,379 B1.

Christie, as discussed above, does not disclose a code page that stores instruction words.

The register file 60 of Christie stores operand values. As recognized by Christie (column 1, line 23, through column 2, line 39), instructions are more than operand values as they also include opcodes (command portions) that Christie's register file 60 does not store.

Yates has been cited by the Patent Office to provide teachings for address fault circuitry, multiplexers, and RISC instructions and does not teach or suggest a code page that stores instruction words.

With respect to all of the claims rejected under 35 USC § 103(a), Applicants respectfully note that they are dependent upon independent claims 1, 22, and 43. As Yates does not cure the deficiencies in Christie described above, claims 5, 17, 19-21, 26, 38 and 40-42 are likewise in condition for allowance.

Response to the Patent Office's Response to Arguments

The Patent Office has presented certain arguments on pages 2-4 of the Final Office Action dated January 29, 2007.

During patent examination, the pending claims must be "given their broadest reasonable interpretation consistent with the specification." MPEP 2111.

Applicant has disclosed, on page 10, lines 7-8, of the specification, as follows: "It should be noted that in a given code page, each 32-bit basic instruction has a corresponding 8-bit extension, or no 32-bit basic instruction has an 8-bit extension."

The Patent Office in pages 2-4 of the Final Office Action has discussed a paging mechanism to retrieve data from the register file that includes the standard register set and the extended register set and has discussed paging hardware that can load into a register via the execution unit. The Patent Office should consider what is being loaded into the register file 60, the Patent Office's asserted analogue to a code page. Being loaded into the register file 60 are operands (see, e.g., Christie, column 8, lines 39-41); instructions are loaded into instruction cache 50 (see, e.g., Christie, column 8, lines 28-37). Please note that Christie distinguishes an instruction as found in instruction cache 50 and decoded by decode unit 76 in column 8, lines 27-37, from an operand as found in register file 60 and requested by decode unit 76. As discussed above, an instruction must have an opcode, i.e., a command portion. An instruction may have a data portion, i.e., the operand.

There are three recitations of "opcode" in the detailed description of Christie. In column

6, lines 47-56, Christie discloses setting and clearing RX bits in the decode control unit 58 in which the use of STX and CLX instructions when REX32 mode is not enabled may result in an "undefined opcode" exception. Christie, column 6, lines 47-56, clearly refers to decode control unit 58 in Figure 5 which is separate from register file 60, also shown in Figure 5. Christie, column 10, lines 45-51, discloses as follows: "Field extension bits 96, 98, and 100 may be used to extend 3-bit fields in an opcode, a Mod R/M byte, and/or a SIB byte of instruction 80 to 4 bits, thus allowing 16 registers to be accessed instead of just 8. Field extension bits 96, 98, and 100 may thus be used to increase the number of general purpose registers in CPU 32 (FIGS. 4 and 5)." Field extension bits 96, 98, and 100 are illustrated in Figure 8. Christie discloses, column 4, lines 22-26, as follows: "FIG. 8 is a diagram of one embodiment of the optional prefix portion of the instruction of FIG. 7, wherein the prefix portion is an extended register prefix byte including an extended register key field, a 64-bit operand size override bit, and three field extension bits." Since the field extension bits 96, 98, and 100 are used to address extended registers, they would not be located within register file 60. Also, the field extension bits 96, 98, and 100 form part of instruction 80 (column 10, lines 45-49) which is shown to be provided to the decode unit 76 and not to the register file in Figure 7. Christie, column 11, lines 16-24, further discloses the field extension bit 100 may be used to extend the Mod R/M byte R/M field, the SIB byte BASE field, or an opcode register reference field from three bits to four, thus forming an extended field.

Clearly, register file 60 does not store instruction words. A code page has instructions. Thus, register file 60 does not correspond to a code page.

The Patent Office asserted on page 2, last line, through page 3, line 2, of the Final Office Action dated January 29, 2007, as follows: "According to column 8, lines 61-67, the execution unit can either output results to the registers or the cache, thus meaning that anything that can be stored within the registers can also be stored within the cache." **The preceding assertion by the Patent Office is misleading.** The Patent Office is referred to Figure 5. Instructions are stored in instruction cache 50. Christie, column 8, lines 61-63, discloses as follows: "Execution unit 78 provides any results of instruction executions either to register file 60 or to data cache 54 (FIG. 5)." A cache is not interchangeable with another cache in Christie. The instruction cache 50 provides instruction words to the decode unit 76 which generates data as a result. This data is then consequently stored in other storage components – i.e., data cache 54 or register file 60

(please see Figure 5).

The Patent Office asserted on page 3, lines 2-11, of the Final Office Action dated January 29, 2007, as follows:

> According to column 8, lines 48-60, the cache provides data to both the execution unit and the paging hardware. Given this, a code page held in cache, provided by the paging hardware, can be loaded into a register via the execution unit. A register can hold data sent from cache. If the extended registers are to be used, useful data must be loaded into the register, as registers cannot generate data. Once a register is done with its data, it can be written back into main memory. Therefore there must exist a paging mechanism to retrieve data from the register file. The existence of registers with extensions and page mechanism along with common usage of registers and cache teaches the limitations of claim 1.

The Patent Office comments as to paging mechanism functionality, extensions, and the form of an extended register from page 3, line 12, through page 4, line 4, of the Final Office Action dated January 29, 2007.

The three independent claims – claim 1, claim 22, and claim 43 - recite "instruction words" and a "code page." Christie discloses neither. "Instruction words" and a "code page" would not be inherent features of the register file 60 of Christie. In fact, Christie is careful to distinguish the instruction cache 50 "instructions" from the operands stored in the register file 60.

Thus, Christie is not an anticipatory reference for claims 1-44.

The Examiner is respectfully requested to reconsider and remove the rejections of the claims, and to allow all of the pending claims 1-44 as now presented for examination. An early notification of the allowability of the pending claims is earnestly solicited.

Respectfully submitted:

Walter J. Malinowski May 18, 2007

Walter J. Malinowski Date

Reg. No.: 43,423

Customer No.: 29683

HARRINGTON & SMITH, PC

4 Research Drive

Shelton, CT 06484-6212

Telephone: (203)925-9400, extension 19

Facsimile: (203)944-0245

email: wmalinowski@hspatent.com

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, P.O. BOX 1450, Alexandria, VA 22313-1450.

_____ _____

Date Name of Person Making Deposit